Computer Physics Communications 200 (2016) 37–43

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

Extending the range of real time density matrix renormalization group simulations

D.M. Kennes [a], C. Karrasch [b,c,*]

[a] Institut für Theorie der Statistischen Physik, RWTH Aachen University and JARA–Fundamentals of Future Information Technology, 52056 Aachen, Germany

[b] Department of Physics, University of California, Berkeley, CA 95720, USA

[c] Materials Sciences Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

ARTICLE INFO

Article history:
Received 13 June 2015
Received in revised form
5 September 2015
Accepted 28 October 2015
Available online 11 November 2015

Keywords:
Density matrix renormalization group
Hints for beginners
Python code

ABSTRACT

We discuss a few simple modifications to time-dependent density matrix renormalization group (DMRG) algorithms which allow to access larger time scales. We specifically aim at beginners and present practical aspects of how to implement these modifications within any standard matrix product state (MPS) based formulation of the method. Most importantly, we show how to 'combine' the Schrödinger and Heisenberg time evolutions of arbitrary pure states $|\psi\rangle$ and operators A in the evaluation of $\langle A\rangle_\psi(t) = \langle\psi|A(t)|\psi\rangle$. This includes quantum quenches. The generalization to (non-)thermal mixed state dynamics $\langle A\rangle_\rho(t) = \mathrm{Tr}[\rho A(t)]$ induced by an initial density matrix ρ is straightforward. In the context of linear response (ground state or finite temperature $T > 0$) correlation functions, one can extend the simulation time by a factor of two by 'exploiting time translation invariance', which is efficiently implementable within MPS DMRG. We present a simple analytic argument for why a recently-introduced disentangler succeeds in reducing the effort of time-dependent simulations at $T > 0$. Finally, we advocate the python programming language as an elegant option for beginners to set up a DMRG code.

1. Introduction

The density matrix renormalization group (DMRG) [1,2] was originally devised [3,4] as a tool to accurately determine *static* ground state properties of one dimensional systems. From a modern perspective, the core DMRG algorithm can be formulated elegantly if one introduces the concept of matrix product state (MPS) [5–8]. In principle, any given state of a 1d system can be expressed as a MPS:

$$|\psi\rangle = \sum_{\{\sigma_l\}} c_{\sigma_1\ldots\sigma_L}|\sigma_1\sigma_2\ldots\sigma_L\rangle \overset{\mathrm{SVD}}{=} \sum_{\{\sigma_l\}} M^{\sigma_1}\cdot M^{\sigma_2}\cdots M^{\sigma_L}|\sigma_1\sigma_2\ldots\sigma_L\rangle, \qquad (1)$$

where σ_l denote single-particle quantum numbers (e.g., $\sigma_l = \uparrow, \downarrow$ for a spin-1/2 system). The mathematical algorithm which allows to obtain the matrices M^{σ_l} from the coefficients $c_{\sigma_1\ldots\sigma_l}$ is the so-called singular value decomposition (SVD). For an arbitrary 1d state, the dimension χ of the matrices (which is often called bond dimension) grows exponentially with the size L of the system. The idea of a ground state DMRG calculation is to fix χ to a value much smaller than e^L and to determine the matrices M^{σ_l} variationally. The reason why this is a reasonable approximation invokes the notion of entanglement. If one cuts the system and the amount of entanglement between left and right half is given by S, then the matrix at the cut site needs to be of dimension $\chi \sim e^S$ to encode this. The entanglement, however, is not an extensive quantity, $S \neq S(L)$, but follows an area law—and the area between two 1d chains is a constant [9,10]. In other words, the ground state of one-dimensional systems[1] is entangled only locally, and this is perfectly coded by a MPS. Finally, the notion of a matrix product state can be generalized to operators so that all 'ingredients' for DMRG calculations are elegantly expressed using the same concepts [11–17]. One can show that each operator A can be recast as a

---

* Corresponding author at: Department of Physics, University of California, Berkeley, CA 95720, USA. Tel.: +1 5106647052.
E-mail address: c.karrasch@fu-berlin.de (C. Karrasch).

[1] More precisely, of gapped 1d systems governed by local Hamiltonians.

http://dx.doi.org/10.1016/j.cpc.2015.10.019
0010-4655/

matrix product operator (MPO)

$$A = \sum_{\{\sigma_l\}} \sum_{\{\tilde{\sigma}_l\}} A^{\sigma_1,\tilde{\sigma}_1} \cdot A^{\sigma_2,\tilde{\sigma}_2} \cdots A^{\sigma_L,\tilde{\sigma}_L} |\{\sigma_l\}\rangle\langle\{\tilde{\sigma}_l\}| \tag{2}$$

if one again allows for an exponentially large bond dimension $\chi \sim e^L$ [2]. Fortunately, most physical observables have a representation in terms of a MPO with a small $\chi = O(1)$ that can be obtained by mere inspection; e.g. purely local operators $A_l = \mathbb{1} \cdots \mathbb{1} \cdot A_l \cdot \mathbb{1} \cdots \mathbb{1}$ trivially are a MPO with $\chi = 1$ (we will discuss a more complex example in Section 3.3).

Time evolution—Following the invention of the ground state DMRG [3,4], several algorithms were developed to simulate the real time evolution [18–23]. This allows to compute correlation functions $\langle \mathrm{gs}|A(t)B|\mathrm{gs}\rangle$[2] or to calculate the evolution of observables $\langle\psi|A(t)|\psi\rangle$ in arbitrary initial states. One of the most common approaches is the so-called time-dependent DMRG, which can be formulated very elegantly using matrix product states. Its basic idea is to split up the time evolution into small steps, $\exp(-iHt) = \exp(-iH\Delta t)\exp(-iH\Delta t)\ldots$ and to Trotter-decompose [32] the exponentials via $\exp(-iH\Delta t) = \exp(-ih_a\Delta t)\exp(-ih_b\Delta t) + O([\Delta t]^2)$[3] into factors which only contain local, mutually commuting terms $h_{a,b}$ such as nearest-neighbor Heisenberg spin interactions $\vec{S}_1\vec{S}_2$. Local terms, however, can be applied straightforwardly to a MPS—they act only on the matrices living at the corresponding sites, e.g.:

$$M^{\sigma_1}, M^{\sigma_2} \overset{\exp(-i\vec{S}_1\vec{S}_2\Delta t)}{\longrightarrow} f(M^{\sigma_1}, M^{\sigma_2}) \overset{\mathrm{SVD}}{\longrightarrow} \tilde{M}^{\sigma_1}, \tilde{M}^{\sigma_2}. \tag{3}$$

The main tool to bring the resulting state back to the form of a MPS is again the singular value decomposition. The algorithm is explained in great detail in the literature [2] and will also be made more explicit in Section 4.

In order to approximate $\langle \mathrm{gs}|A(t)B|\mathrm{gs}\rangle$ or $\langle\psi|A(t)|\psi\rangle$ to a certain accuracy, the dimension of the matrix product state needs be increased with time, often even exponentially. Physically, this is due to the buildup of entanglement in the system, and it limits the accessible time scales. Modified algorithms such as transverse folding [33] hold the promise of substantially extending the range of simulations, but implementing them in practice requires some effort (for a comprehensive overview of other available approaches see Ref. [2] and references therein). It is one of the goals of this paper to discuss a few *simple* 'recipes' that allow to reach larger times in DMRG calculations. In particular, we will investigate if the Schrödinger and Heisenberg pictures of quantum mechanics can be combined efficiently.

Finite temperatures—Standard DMRG methods allow computing the time evolution of a *pure* state and are thus not directly applicable at finite temperatures. In order to simulate dynamics at $T > 0$, one can use operator space DMRG [34,35], or – equivalent mathematically – one can express the thermal statistical operator $\rho_T \sim e^{-H/T}$ as a partial trace over a pure state $|\Psi_T\rangle$ living in an enlarged Hilbert space where auxiliary degrees of freedom Q encode the thermal bath [36–40][4]:

$$\rho_T = \mathrm{Tr}_Q |\Psi_T\rangle\langle\Psi_T|. \tag{4}$$

A finite-T correlation function can in principle be obtained straightforwardly by carrying out real- and imaginary time evolutions of the state $|\psi_\infty\rangle$ which purifies ρ_T at $T = \infty$ [2]:

$$\begin{aligned} C_T^{AB}(t) &= \mathrm{Tr}\left[\rho_T A(t) B\right] \\ &= \langle\psi_T|A(t)B|\psi_T\rangle \sim \langle\psi_\infty|e^{-H/2T}A(t)Be^{-H/2T}|\psi_\infty\rangle. \end{aligned} \tag{5}$$

Since $\rho_\infty \sim 1$, the state $|\psi_\infty\rangle$ is known; it is given by a product of local states, which for a spin-1/2 system read $|\psi_\infty\rangle \sim |\uparrow\downarrow_Q\rangle - |\downarrow\uparrow_Q\rangle$. It is another goal of this paper to recapitulate recent ideas [40,34] of how to extend the range of real-time DMRG simulation at finite temperature in a didactic way.

Structure of the paper—We will discuss a few *simple* 'recipes' that allow to reach larger times in DMRG calculations. The combination of the Schrödinger and Heisenberg pictures is investigated in Section 2. Prior ideas of how to extend the range of finite temperature calculations are presented in a didactical way in Section 3. We specifically aim at an audience of beginners and ask: Assuming that one has a standard MPS based DMRG code at hand, what are the most important practical steps necessary to incorporate these recipes? We will particularly elaborate how to incorporate Abelian symmetries [48]. For colleagues new to the realm of DMRG, we try to advocate the method in general by showing how straightforwardly its core algorithms can be implemented within the python programming language (Section 4). Throughout this paper, we perform our DMRG calculations using a fixed small discarded weight and a 4th order Trotter decomposition of the time evolution operators.

2. Schrödinger vs. Heisenberg picture

2.1. General aspects

Let us assume we want to compute the time evolution of an observable A in a given state $|\psi\rangle$,

$$\langle A\rangle_\psi(t) = \langle\psi|A(t)|\psi\rangle, \tag{6}$$

or similarly at finite temperature by enlarging the Hilbert space and choosing $|\psi\rangle$ as the state which purifies the initial density matrix, $\langle A\rangle_\rho(t) = \mathrm{Tr}\left[\rho A(t)\right] = \langle\psi_\rho|A(t)|\psi_\rho\rangle$. The most straightforward way to evaluate Eq. (6) within DMRG is to simulate $e^{-iHt}|\psi\rangle$. This corresponds to a time evolution in the Schrödinger picture. If one expresses A as a matrix product operator, one can alternatively switch to the Heisenberg picture and calculate the operator time evolution $e^{iHt}Ae^{-iHt}$ [11–13,48,14–17]. This is equivalent mathematically but different algorithmically since $e^{iHt}Ae^{-iHt}$ might have a more efficient representation in terms of a MPO than $e^{-iHt}|\psi\rangle$ has in terms of a MPS (or vice versa). For example, the Schrödinger picture is certainly advantageous whenever $|\psi\rangle$ is close to an eigenstate of H. Of course, it is possible to simply split the time evolution between the Schrödinger- and Heisenberg picture,

$$\langle A\rangle_\psi(2t) = \langle\psi|e^{iHt}A(t)e^{-iHt}|\psi\rangle, \tag{7}$$

and evaluate $e^{-iHt}|\psi\rangle$ as well as $A(t)$ individually. Generally speaking, if one stops each simulation at times t_ψ and t_A where a fixed bond dimension is reached, one can *always* reach larger time scales $t_\psi + t_A > t_\psi, t_A$.[5] In practice, however, t_ψ might be significantly smaller than t_A or vice versa, and the additional effort to implement Eq. (7) might simply not be justified. We will now discuss this using an instructive example.

2.2. Examples

We consider the time evolution of a local spin operator S_l^z under the XXZ spin chain Hamiltonian:

$$H = \sum_{l=1}^{L-1}\left[\frac{1}{2}\left(S_l^+S_{l+1}^- + S_l^-S_{l+1}^+\right) + \Delta S_l^zS_{l+1}^z\right] + b\sum_{l=1}^{L}S_l^z, \tag{8}$$

---

[2] For other approaches to calculate correlation functions, see Refs. [1,24–31].

[3] In practice, higher-order approximations are used; see, e.g., Ref. [2].

[4] Various other ways to incorporate finite temperatures within DMRG, e.g., using minimally entangled typical thermal states [41], can be found in Refs. [42–47].

[5] This is not hindered by a more costly 'overlap' calculation; see Section 2.3 for details.